24/PRTS

09/719051

JC01 Rec'd PCT/PTO 07 DEC 2000

DESCRIPTION

IMAGE PROCESSING METHOD AND IMAGE PROCESSING APPARATUS

TECHNICAL FIELD

The present invention relates to image processing technology utilizing resolution conversion for a video signal that has been encoded by a high-efficiency compression coding method complying with the MPEG-2, for example.

BACKGROUND ART

In recent years, the MPEG-2 (Moving Picture Experts Group-2) standard has been widely used as a high-efficiency compression coding method applicable to transmitting or recording a video signal. According to the MPEG-2 standard, video frames (or fields) are encoded by classifying them into the three picture types of I-, P- and B-pictures. Intra-frame coding is performed on an I-picture. Inter-frame forward predictive coding is performed on a P-picture by using an I- or P-picture, which is ahead of a current picture on the time axis, as a reference frame. And inter-frame bidirectionally predictive coding is performed on a B-picture by using I- and/or P-pictures, which are ahead of and behind the current picture on the time axis, as respective reference frames.

Motion compensation is utilized in predictively coding P-

and B-pictures. According to the MEPG-2, motion compensation is performed on a so-called "macroblock" (consisting of 16×16 pixels) basis. In general, a block matching technique, by which a block with the highest correlation is found in a reference frame, is used in estimating a motion vector for the motion compensation purposes.

Hereinafter, known image processing techniques utilizing resolution conversion will be described.

FIG. **22** is a block diagram for a known image processing apparatus that performs MPEG-2 decoding, resolution conversion and MPEG-2 encoding. In FIG. **22**, an encoded stream **CSA** is decoded by an MPEG-2 decoder **510** to be an original video signal **SHR** with a high resolution. The original video signal **SHR** is converted by a resolution converter **520** into a new video signal **SLR** with a low resolution. The new video signal **SLR** is encoded by an MPEG-2 encoder **530** in compliance with the MEPG-2 so as to be output as an encoded stream **CSB**.

If the original video signal **SHR** is an interlaced signal, then the resolution converter **520** usually performs the conversion on a field-by-field basis. This is because if the resolution conversion is performed on a frame structure as a whole, then the resolution will increase in a still-picture portion of the frame as compared to the resolution performed on a field structure, but a moving-picture portion thereof cannot be converted appropriately.

FIG. 23 illustrates how resolution conversion may be performed on a pair of fields without changing the structure thereof. As shown in FIG. 23, a first field of a new video signal is generated from a first field of an original video signal, and a second field of the new video signal is generated from a second field of the original video signal.

According to another proposed method, motion estimation is carried out before resolution conversion to find still- and moving-picture portions and then the resolution conversion is performed on the still- and moving-picture portions using the frame and field structures, respectively.

FIG. 24 illustrates two methods of resolution conversion. In FIG. 24, (a) illustrates an image corresponding to an original video signal with a high resolution; (b) illustrates a letterbox image with a low resolution obtained by converting the original video signal shown in (a); and (c) illustrates a squeeze image with a low resolution obtained by converting the original video signal shown in (a). In FIG. 24(a), the aspect ratio of the original video signal is supposed to be 16:9. The aspect ratio of the new video signals is supposed to be 4:3. As for the letterbox image shown in FIG. 24(b), the entire original image has been downscaled at the same ratio both vertically and horizontally and data in the shape of black bands has been added to the top and bottom of the image. As for the squeeze image shown in FIG. 24(c) on the other hand,

the original image has been downscaled at a ratio vertically and at a different ratio horizontally. That is to say, the aspect ratio of the image shown in FIG. **24(c)** is 4:3.

Problems to be Solved

The prior art described above, however, has the following drawbacks.

Firstly, if motion estimation is performed before resolution conversion, then a huge amount of processing is needed for the motion estimation, thus increasing the amount of hardware or software required.

Secondly, as for the configuration shown in FIG. **22**, the MPEG-2 encoder **530** should perform normal MPEG-2 encoding processing completely to generate the encoded stream **CSB** that is just different in resolution from the encoded stream **CSA**. Thus, the amount of processing will also increase, so the amount of hardware or software required rises, too.

Furthermore, no matter whether the new video signal **SLR** should be represented as a letterbox image or squeeze image, the MPEG-2 encoder **530** always applies the same encoding method thereto and cannot make full use of the characteristics of a particular resolution conversion method.

DISCLOSURE OF INVENTION

An object of the present invention is realizing image

processing utilizing resolution conversion with the amount of processing reduced.

Specifically, an inventive image processing method includes: a decoding step, in which an encoded stream, obtained by encoding an original video signal with a first resolution, is decoded and in which an encoding parameter is extracted from the encoded stream; and a resolution converting step, in which a characteristic of the original video signal is identified by the encoding parameter and in which the original video signal decoded is converted into a new video signal with a second resolution by a resolution conversion method associated with the characteristic.

And in the resolution converting step, a motion characteristic of a picture represented by the original video signal is preferably identified as the characteristic of the original video signal. Furthermore, the encoding parameter preferably includes at least one of: a motion vector representing a quantity of motion of a video unit; a type of orthogonal transformation to be performed using either a frame structure or a field structure; and a mode of motion compensation to be performed using either the frame structure or the field structure.

Also, in the resolution converting step, a picture represented by the original video signal decoded is preferably divided into a still area and a moving area by using the en-

coding parameter, and the original video signal is preferably converted into the new video signal with mutually different resolution conversion methods applied to the still and moving areas. And the original video signal is preferably an interlaced signal. The resolution conversion is preferably performed on the still area on a frame basis, while the resolution conversion is preferably performed on the moving area on a field basis. Furthermore, the encoding parameter is preferably a motion vector representing a quantity of motion of a video unit, and the area division is preferably carried out based on a result of comparison between the absolute value of the motion vector and a predetermined value.

An inventive image processing apparatus includes: a video decoder for decoding an encoded stream, obtained by encoding an original video signal with a first resolution, and for extracting an encoding parameter from the encoded stream; and a resolution converter, which receives the original video signal and the encoding parameter that have been output from the video decoder, identifies a characteristic of the original video signal by the encoding parameter and converts the original video signal into a new video signal with a second resolution by a resolution conversion method associated with the characteristic.

And the resolution converter preferably identifies a motion characteristic of a picture represented by the original

video signal as the characteristic of the original video signal. Furthermore, the encoding parameter preferably includes at least one of: a motion vector representing a quantity of motion of a video unit; a type of orthogonal transformation to be performed using either a frame structure or a field structure; and a mode of motion compensation to be performed using either the frame structure or the field structure.

Also, the resolution converter preferably includes: an area dividing section for dividing a picture represented by the input original video signal into a still area and a moving area by using the encoding parameter; a still-area resolution converting section for converting a video signal representing the still area, which has been output from the area dividing section, into the video signal with the second resolution; and a moving-area resolution converting section for converting a video signal representing the moving area, which has been output from the area dividing section, into the video signal with the second resolution. Furthermore, the original video signal is preferably an interlaced signal, the still-area resolution converting section preferably performs the resolution conversion on a frame basis, and the moving-area resolution converting section preferably performs the resolution conversion on a field basis. Furthermore, the encoding parameter is preferably a motion vector representing a quantity of motion of a video unit, and the area dividing

section preferably performs the area division based on a result of comparison between the absolute value of the motion vector and a predetermined value.

Another inventive image processing method includes: a decoding step, in which an encoded stream, obtained by encoding an original video signal with a first resolution, is decoded and in which a motion vector is extracted from the encoded stream; and a resolution converting step, in which the original video signal decoded is converted into a new video signal with a second resolution by using the motion vector extracted.

And the resolution converting step preferably includes an area dividing step, in which a picture represented by the original video signal decoded is divided into a quasi-still area and a moving area by using the motion vector extracted. The resolution is preferably converted into that of the new video signal by using the extracted motion vector for the quasi-still area and without using the extracted motion vector for the moving area.

And in the area dividing step, pixel-by-pixel motion vectors with directions similar to that of the extracted motion vector are preferably estimated from the extracted motion vector. An area, in which the pixel-by-pixel motion vectors have been estimated, is preferably regarded as the quasi-still area, while an area, in which none of the vectors

has been estimated, is preferably regarded as the moving area. Furthermore, in the quasi-still area, the resolution is preferably converted into that of the new video signal by using the pixel-by-pixel motion vectors estimated.

Also, in the area dividing step, an area, in which the absolute value of the extracted motion vector is less than a predetermined threshold value, is preferably regarded as the quasi-still area. On the other hand, an area, in which the absolute value is greater than the threshold value, is preferably regarded as the moving area.

Another inventive image processing apparatus includes: a video decoder for decoding an encoded stream, obtained by encoding an original video signal with a first resolution, and for extracting a motion vector from the encoded stream; and a resolution converter, which receives the original video signal and the motion vector that have been output from the video decoder and converts the original video signal into a new video signal with a second resolution by using the motion vector.

And the resolution converter preferably includes: an area dividing section, which receives the original video signal and the motion vector and divides a picture represented by the original video signal into a quasi-still area and a moving area by using the motion vector; a quasi-still-area resolution converting section for converting a video signal representing the quasi-still area, which has been output from the area di-

viding section, into the video signal with the second resolution by using the motion vector; and a moving-area resolution converting section for converting a video signal representing the moving area, which has been output from the area dividing section, into the video signal with the second resolution without using the motion vector.

Also, the area dividing section preferably includes a motion vector estimating section for estimating, from the motion vector, pixel-by-pixel motion vectors with directions similar to that of the motion vector. An area, in which the pixel-by-pixel motion vectors have been estimated by the motion vector estimating section, is preferably regarded as the quasi-still area, while an area, in which none of the vectors has been estimated, is preferably regarded as the moving area. Furthermore, the quasi-still-area resolution converting section preferably converts the resolution by using the pixel-by-pixel motion vectors estimated by the motion vector estimating section.

Moreover, the area dividing section preferably regards an area, in which the absolute value of the motion vector is less than a predetermined threshold value, as the quasi-still area, and an area, in which the absolute value is greater than the threshold value, as the moving area.

Still another inventive image processing method includes the steps of: decoding a first encoded stream, obtained by en-

coding an original video signal with a first resolution, and extracting a first encoding parameter from the first encoded stream; converting the original video signal decoded into a new video signal with a second resolution; changing the first encoding parameter into a second encoding parameter for use in encoding the new video signal; and encoding the new video signal using the second encoding parameter, thereby generating a second encoded stream.

And in the step of changing the encoding parameters, the first encoding parameter, which has been used to encode a first area of a picture represented by the original video signal, is preferably changed into the second encoding parameter, which will be used to encode a second area of a picture represented by the new video signal. The second area preferably includes part of the picture represented in the first area.

And the first and second encoding parameters are preferably motion vectors. Furthermore, in the step of changing the encoding parameters, a value (e.g., a weighted average), which has been obtained by performing a predetermined arithmetic operation on the motion vectors of the first area, is preferably regarded as the motion vector of the second area.

Also, the first and second encoding parameters preferably each represent a type of orthogonal transformation to be performed using either a frame structure or a field struc-

ture.

Still another inventive image processing apparatus includes: a video decoder for decoding a first encoded stream, obtained by encoding an original video signal with a first resolution, and for outputting a first encoding parameter from the first encoded stream; a resolution converter for converting the original video signal, which has been output from the video decoder, into a new video signal with a second resolution; an encoding parameter changer for changing the first encoding parameter, which has been output from the video decoder, into a second encoding parameter for use in encoding the new video signal; and a video encoder for encoding the new video signal output from the resolution converter by using the second encoding parameter output from the encoding parameter changer, thereby generating a second encoded stream.

And the encoding parameter changer preferably changes the first encoding parameter, which has been used to encode a first area of a picture represented by the original video signal, into the second encoding parameter, which will be used to encode a second area of a picture represented by the new video signal. The second area preferably includes part of the picture represented in the first area.

And the first and second encoding parameters are preferably motion vectors. Furthermore, the encoding parameter changer preferably regards a value (e.g., a weighted average)

which has been obtained by performing a predetermined arithmetic operation on the motion vectors of the first area, as the motion vector of the second area.

Also, the first and second encoding parameters preferably each represent a type of orthogonal transformation to be performed using either a frame structure or a field structure.

Yet another inventive image processing method includes the steps of: decoding a first encoded stream, obtained by encoding an original video signal with a first resolution, and extracting a first motion vector from the first encoded stream; converting the original video signal decoded into a new video signal with a second resolution; defining setting information based on the first motion vector to estimate a second motion vector for use in encoding the new video signal; estimating the second motion vector in accordance with the setting information defined; and encoding the new video signal using the second motion vector estimated, thereby generating a second encoded stream.

And an initial value of the second motion vector is preferably determined as the setting information. Alternatively, a search range in which the second motion vector is estimated is preferably determined as the setting information.

Yet another inventive image processing apparatus in-

cludes: a video decoder for decoding a first encoded stream, obtained by encoding an original video signal with a first resolution, and for extracting a first motion vector from the first encoded stream; a resolution converter for converting the original video signal, which has been output from the video decoder, into a new video signal with a second resolution; a motion compensation setter for generating setting information based on the first motion vector, which has been output from the video decoder, to estimate a second motion vector for use in encoding the new video signal; and a video encoder for estimating the second motion vector in accordance with the setting information generated by the motion compensation setter and encoding the new video signal, which has been output from the resolution converter, using the second motion vector estimated, thereby generating a second encoded stream.

And the motion compensation setter preferably determines an initial value of the second motion vector as the setting information. Alternatively, the motion compensation setter preferably determines a search range, in which the second motion vector is estimated, as the setting information.

Yet another inventive image processing method includes: converting an original video signal having a first resolution into a new video signal having a second resolution and a black-level area in part of its picture; encoding the new video signal except the black-level area thereof, thereby

generating a first encoded stream; and combining a second encoded stream, obtained by encoding the black-level area of the video signal, with the first encoded stream, thereby generating an encoded stream for the new video signal.

An alternative inventive image processing method includes: converting an original video signal having a first resolution into part of a new video signal, the new video signal having a second resolution and a black-level area in the other part of its picture; encoding the video signal, thereby generating a first encoded stream; and combining a second encoded stream, obtained by encoding the black-level area of the video signal, with the first encoded stream, thereby generating an encoded stream for the new video signal.

Yet another inventive image processing apparatus includes: a resolution converter for converting an original video signal having a first resolution into a new video signal having a second resolution and a black-level area in part of its picture; and a video encoder for encoding the new video signal except the black-level area thereof to generate a first encoded stream and combining a second encoded stream, obtained by encoding the black-level area of the video signal, with the first encoded stream, thereby generating an encoded stream for the new video signal.

An alternative inventive image processing apparatus includes: a resolution converter for converting an original

video signal having a first resolution into part of a new video signal, the new video signal having a second resolution and a black-level area in the other part of its picture; and a video encoder for encoding the video signal to generate a first encoded stream and combining a second encoded stream, obtained by encoding the black-level area of the video signal, with the first encoded stream, thereby generating an encoded stream for the new video signal.

BRIEF DESCRIPTION OF DRAWINGS

FIG. 1 is a block diagram illustrating an overall configuration for an image processing apparatus according to a first embodiment of the present invention.

FIG. 2 is a block diagram illustrating a configuration for the video decoder shown in FIG. 1.

FIG. 3 is a block diagram illustrating a configuration for the resolution converter shown in FIG. 1.

FIGS. 4(a) and 4(b) are schematic representations illustrating how motion may be recognized in the first embodiment of the present invention: FIG. 4(a) illustrates a situation where a motion vector is used as an encoding parameter; and FIG. 4(b) illustrates a situation where a DCT type is used as an encoding parameter.

FIGS. 5(a) through 5(c) are schematic representations illustrating how area division may be carried out according

to the first embodiment of the present invention: FIG. 5(a) illustrates a result of motion recognition; FIG. 5(b) illustrates a still area; and FIG. 5(c) illustrates moving areas.

FIGS. 6(a) and 6(b) are schematic representations illustrating how resolution conversion may be performed according to the first embodiment of the present invention: FIG. 6(a) illustrates a resolution conversion using a frame structure as it is; and FIG. 6(b) illustrates a resolution conversion using field structures as they are.

FIGS. 7(a) through 7(e) are schematic representations illustrating how areas may be combined according to the first embodiment of the present invention.

FIG. 8 is a block diagram illustrating an overall configuration for an image processing apparatus according to a second embodiment of the present invention.

FIG. 9 is a block diagram illustrating a configuration for the resolution converter shown in FIG. 8.

FIG. 10 is a schematic representation illustrating how the motion vector estimating and quasi-still-area resolution converting sections shown in FIG. 9 may operate.

FIG. 11 is a schematic representation illustrating how the moving-area resolution converting section shown in FIG. 9 may operate.

FIG. 12 is a block diagram illustrating another exemplary configuration for the resolution converter shown in

FIG. 8.

FIG. 13 is a block diagram illustrating an overall configuration for an image processing apparatus according to a third embodiment of the present invention.

FIGS. 14(a) through 14(c) are schematic representations illustrating how the encoding parameter changer shown in FIG. 13 may operate.

FIG. 15 is a block diagram illustrating a configuration for the video encoder shown in FIG. 13.

FIG. 16 is a block diagram illustrating an overall configuration for an image processing apparatus according to a fourth embodiment of the present invention.

FIG. 17 is a block diagram illustrating a configuration for the video encoder shown in FIG. 16.

FIGS. 18(a) through 18(c) are schematic representations illustrating how the motion vector calculator shown in FIG. 17 may operate.

FIG. 19 is a block diagram illustrating an overall configuration for an image processing apparatus according to a fifth embodiment of the present invention.

FIGS. 20(a) and 20(b) illustrate an exemplary resolution conversion according to the fifth embodiment of the present invention.

FIG. 21 is a block diagram illustrating a configuration for the video encoder shown in FIG. 19.

FIG. **22** is a block diagram illustrating a configuration for a known image processing apparatus.

FIGS. **23(a)** and **23(b)** are a schematic representation illustrating how resolution conversion may be performed using field structures.

FIGS. **24(a)** through **24(c)** illustrate two types of resolution conversion.

BEST MODE FOR CARRYING OUT THE INVENTION

(Embodiment 1)

FIG. **1** is a block diagram illustrating an overall configuration for an image processing apparatus according to a first embodiment of the present invention. As shown in FIG. **1**, the image processing apparatus according to this embodiment includes video decoder **10** and resolution converter **20**. An encoded stream **CS** of an original video signal with a high resolution (i.e., a first resolution) is input. In response, the video decoder **10** decodes the input encoded stream **CS** into the original video signal **SHR**. Not just this decoding, the decoder **10** also extracts an encoding parameter **PAR** from the encoded stream **CS**. The resolution converter **20** converts the decoded original video signal **SHR** into a new video signal **SLR** with a low resolution (i.e., a second resolution) using the encoding parameter **PAR** and then outputs the new video signal.

The encoded stream **CS** is herein supposed to have been

encoded in compliance with the MPEG-2. Accordingly, the video decoder **10** is a decoder complying with the MPEG-2 (Moving Picture Experts Group 2) standard. Also, both the original and new video signals **SHR** and **SLR** are herein supposed to be interlaced signals (i.e., interlaced scanning signals).

FIG. **2** is a diagram illustrating an internal configuration for the video decoder **10**. As shown in FIG. **2**, the video decoder **10** includes variable-length decoding section **11**, inverse quantizing section **12**, inverse DCT (discrete cosine transform) section **13**, frame memory **14**, system control section **15**, adder **16** and switch **17**. The variable-length decoding section **11** also extracts the encoding parameter **PAR**.

Hereinafter, it will be described how the video decoder **10** shown in FIG. **2** operates. The encoded stream **CS** representing one frame has been encoded on a macroblock (consisting of 16 × 16 pixels) basis. And the macroblocks of the encoded stream **CS** are sequentially input to the video decoder **10** one after another.

The input encoded stream **CS** is variable-length decoded by the variable-length decoding section **11**. Part of the output of the variable-length decoding section **11**, which is associated with picture data, is input to the inverse quantizing section **12**, while the remaining data other than the picture data is sent to the system control section **15**. The data delivered from the variable-length decoding section **11** to the

system control section **15** includes picture types, DCT types and motion vectors, which are encoding parameters of the encoded stream **CS**. As used herein, the "picture type" represents whether a frame has been intra-frame coded or inter-frame coded, while the "DCT type" represents whether the structure, with which a block has been subjected to the DCT process, is a field structure or a frame structure.

The data that has been input to the inverse quantizing section **12** is inversely quantized and then passed to the inverse DCT section **13**. The inverse DCT section **13** performs an inverse DCT operation on the data. Herein, where the MPEG-2 standard is complied with, the DCT process is carried out on an 8×8 pixel block basis. If the data has been intra-frame coded, then the switch **17** is connected to a. On the other hand, if the data has been inter-frame coded, then the switch **17** is connected to b. Supposing the encoded stream **CS** of an inter-frame coded frame is now being processed, the switch **17** will be connected to b. Thus, the output of the inverse DCT section **12** is output to the adder **16**.

Using the motion vector that has been sent from the variable-length decoding section **11** to the system control section **15**, a reference picture is retrieved from the frame memory **14**. A frame to be used as the reference picture has already been decoded and stored on the frame memory **14**. The data on which the inverse DCT has been performed (i.e., the

output of the inverse DCT section **13**) and the reference picture (i.e., the data that has been read out from the frame memory **14**) are added together at the adder **16**. And the sum is stored as a decoded picture on the frame memory **14**. In this manner, the macroblocks are sequentially decoded and stored on the frame memory **14** one after another.

Decoding will also be performed in the same way on the succeeding frames and the results will be sequentially stored on the frame memory **14**. It should be noted that if the frame has been intra-frame coded, then the switch **17** will be connected to a so that the output of the inverse DCT section **13** will be stored on the frame memory **14** as it is. Alternatively, if the frame has been inter-frame coded, then the switch **17** will be connected to b. In that case, the output of the inverse DCT section **13** and the reference picture will be added together by the adder **16** and the sum will be stored on the frame memory **14**.

The decoded pictures, which have been stored on the frame memory **14** this way, are sequentially output as the original video signal **SHR** with time from the frame memory **14**. The encoding parameter **PAR** is also output from the system control section **15**.

FIG. **3** is a diagram illustrating an internal configuration for the resolution converter **20**. As shown in FIG. **3**, the resolution converter **20** includes area dividing section

21, still-area resolution converting section **23**, moving-area resolution converting section **24** and area combining section **25**. The area dividing section **21** includes picture dividing section **21a** and motion recognizing section **21b**.

Hereinafter, it will be described how the resolution converter **20** shown in FIG. **3** operates. First, the motion recognizing section **21b** recognizes the motion characteristic of the picture, represented by the original video signal **SHR**, using the encoding parameter **PAR** received. Then, the motion recognizing section **21b** distinguishes still and moving areas from each other.

FIG. **4** illustrates how the motion recognizing section **21b** recognizes motion. In FIG. **4**, **(a)** illustrates an exemplary motion recognition where a motion vector is used as the encoding parameter **PAR**, while **(b)** illustrates another exemplary motion recognition where a DCT type is used as the encoding parameter **PAR**.

First, it will be described with reference to FIG. **4(a)** how the motion may be recognized where a motion vector is used as the encoding parameter **PAR**. In FIG. **4(a)**, a frame picture has been divided into multiple macroblocks, each of which is used as a video unit consisting of 16 × 16 pixels. According to the MPEG-2 standard, a motion vector represents the horizontal and vertical displacement of a macroblock at a step size of 0.5 pixel. In this case, the motion vector is

compared to a predetermined value and the motion is recognized based on a result of the comparison. For example, if the absolute value of the motion vector is less than the predetermined value, then the macroblock in question is regarded as belonging to the still area. Conversely, if the absolute value is greater than the predetermined value, then the macroblock in question is regarded as belonging to the moving area. In FIG. **4(a)**, the solid black macroblocks belong to the moving areas, while the open macroblocks belong to the still area.

Next, it will be described with reference to FIG. **4(b)** how the motion may be recognized where a DCT type is used as the encoding parameter **PAR**. In FIG. **4(b)**, a frame picture has been divided into multiple blocks, each consisting of 8 × 8 pixels. According to the MPEG-2 standard, DCT is performed on each block using either a frame structure or a field structure to obtain the smaller sum of differences between vertically adjacent pixels. As used herein, the DCT type means the type of the structure the DCT operation has been performed with. In the illustrated example, if the DCT type is the frame structure, then the block in question is regarded as belonging to the still area. Conversely, if the DCT type is the field structure, then the block in question is regarded as belonging to the moving area. In FIG. **4(b)**, the solid black blocks belong to the moving areas, while the open

blocks belong to the still area.

The motion recognizing section **21b** performs the motion recognition in this manner and passes its result to the picture dividing section **21a**. In accordance with the result of motion recognition obtained from the motion recognizing section **21b**, the picture dividing section **21a** divides a frame picture into the still and moving areas.

FIG. **5** illustrates schematic representations showing an exemplary area division. In FIG. **5**, **(a)** illustrates the result of motion recognition shown in FIG. **4(a)**; **(b)** illustrates a still area that has been extracted by the area dividing section **21**; and **(c)** illustrates moving areas extracted by the area dividing section **21**.

Part of the video data represented by the original video signal **SHR**, which has been classified as the still area, is input to the still-area resolution converting section **23**. The other part of the video data, which has been classified as the moving area, is input to the moving-area resolution converting section **24**. The still- and moving-area resolution converting sections **23** and **24** perform the resolution conversion by respective methods associated with the input video data.

FIG. **6** illustrates how the resolution of the original video signal **SHR** may be converted into that of the new video signal **SLR** in the still- and moving-area resolution convert-

ing sections **23** and **24**. In FIG. **6**, columns of pixels vertically arranged are illustrated. Specifically, "○" represents a pixel belonging to a first field, while "△" represents a pixel belonging to a second field. In the example illustrated in FIG. **6**, the vertical pixels are reduced by 1/2. In FIG. **6**, **(a)** illustrates a resolution conversion using a frame structure as it is, while **(b)** illustrates a resolution conversion using field structures as they are.

The still-area resolution converting section **23** performs the resolution conversion using the frame structure as it is. Specifically, as shown in FIG. **6(a)**, the video data of the first field for the new video signal **SLR** is generated by using pixels belonging to both the first and second fields of the original video signal **SHR**. The video data of the second field for the new video signal **SLR** is also generated by using pixels belonging to both the first and second fields of the original video signal **SHR**.

On the other hand, the moving-area resolution converting section **24** performs the resolution conversion using the field structures as they are. Specifically, as shown in FIG. **6(b)**, the video data of the first field for the new video signal **SLR** is generated only from the pixels belonging to the first field of the original video signal **SHR**. And the video data of the second field for the new video signal **SLR** is generated only from the pixels belonging to the second field of the

original video signal **SHR**.

The new video signals, which have been generated by the still- and moving-area resolution converting sections **23** and **24** to represent the still and moving areas, respectively, are output to the area combining section **25**. The area combining section **25** combines these new video signals for the still and moving areas with each other, thereby converting the resultant signal into a frame picture and outputting it as a new video signal **SLR**.

FIG. **7** illustrates how the area combining section **25** may operate. In FIG. **7**, **(a)** and **(b)** illustrate the pictures input to the still- and moving-area resolution converting sections **23** and **24**, respectively, i.e., the pictures shown in FIGS. **5(b)** and **5(c)**. FIGS. **7(c)** and **7(d)** illustrate the pictures output from the still- and moving-area resolution converting sections **23** and **24**, respectively. The number of pixels has been cut down as a result of the resolution conversion. And the area combining section **25** produces a combined picture such as that illustrated in FIG. **7(e)**, i.e., the new video signal **SLR**, from the pictures shown in FIGS. **7(c)** and **7(d)**.

As described above, according to this embodiment, an encoded stream of an original video signal is decoded while an encoding parameter is extracted from the encoded stream. Next, the characteristic of the original video signal is rec-

ognized by the encoding parameter and then the original video signal is converted into a new video signal by a resolution converting method associated with the characteristic. In this manner, the amount of processing, including motion vector calculation for resolution conversion purposes, can be reduced considerably. In addition, a picture, represented by the original video signal, is classified as still and moving areas according to the encoding parameter and the resolution conversion is carried out with mutually different methods applied to the still and moving areas. As a result, a new video signal of quality can be obtained with the amount of processing drastically reduced.

In the foregoing embodiment, motion vectors and DCT types are used as exemplary encoding parameters **PAR** to recognize the motion characteristic of a picture. Alternatively, any other encoding parameter may also be used. Examples of other applicable encoding parameters include motion compensation modes. As used herein, the "motion compensation mode" represents whether the structure with which motion compensation has been performed is a field structure or frame structure. In using the motion compensation modes as the encoding parameters **PAR** of this embodiment, the resolution conversion may be performed using the field structures for an area with a motion compensation mode associated with the field structures. And as for an area with a motion compensation mode associated

with the frame structure, the resolution conversion may be performed using the frame structure.

Also, in this embodiment, the resolution conversion is performed using the frame structure for the still area and the field structures for the moving areas, respectively. Alternatively, any other resolution converting method is also applicable.

Moreover, in performing the resolution conversion using the frame or field structures according to this embodiment, each pixel for the new video signal **SLR** is generated from the values of two vertically adjacent pixels for the original video signal **SHR** as shown in FIGS. **6(a)** and **6(b)**. However, any other combination of pixels may be used instead.

Furthermore, in this embodiment, the resolution is halved. Alternatively, the resolution may be cut down at any other ratio.

Furthermore, in this embodiment, an original video signal with a high resolution is converted into a new video signal with a low resolution. However, even if a low-resolution original video signal should be converted into a high-resolution new video signal conversely, the encoding parameters may also be used as in this embodiment.

(Embodiment 2)

FIG. **8** is a block diagram illustrating an overall con-

figuration for an image processing apparatus according to a second embodiment of the present invention. As shown in FIG. 8, the image processing apparatus according to this embodiment includes video decoder **110** and resolution converter **120**. An encoded stream **CS** of an original video signal with a low resolution (i.e., a first resolution) is input. In response, the video decoder **110** decodes the input encoded stream **CS** into the original video signal **SLR**. Not just this decoding, the decoder **110** also extracts a motion vector **MV** from the encoded stream **CS**. The resolution converter **120** converts the decoded original video signal **SLR** into a new video signal **SHR** with a high resolution (i.e., a second resolution) using the motion vector **MV** and outputs the new video signal.

The original video signal **SLR** is herein supposed to be an interlaced signal (i.e., interlaced scanning signal), while the new video signal **SHR** is herein supposed to be a progressive signal (i.e., progressive scanning signal).

The video decoder **110** has basically the same configuration as the video decoder **10** according to the first embodiment. The only difference between these decoders **110** and **10** is that the decoder **110** outputs the motion vector **MV**. Thus a detailed description thereof will be omitted herein.

FIG. **9** is a block diagram illustrating an internal configuration for the resolution converter **120**. As shown in FIG. **9**, the resolution converter **120** includes area dividing

section **121**, quasi-still-area resolution converting section **123**, moving-area resolution converting section **124**, area combining section **125** and frame memory **127**. Also, the area dividing section **121** includes motion vector estimating section **121a**, motion recognizing section **121b** and picture dividing section **121c**.

The frame memory **127** stores the input original video signals **SLR** thereon. Using the motion vector **MV** output from the video decoder **110**, a current field of the original video signal **SLR** and preceding fields thereof stored on the frame memory **127**, the motion vector estimating section **121a** estimates a pixel-by-pixel motion vector **MV**α, which has a direction similar to that of the motion vector **MV**, for the current field and another field temporally close to the current field.

Hereinafter, it will be described with reference to FIG. **10** how the motion vector estimating section **121a** operates. FIG. **10** schematically illustrates pixels of three frames **n**, **n+1** and **n+2** of the original video signal **SLR**. In FIG. **10**, "○" represents a pixel belonging to the first field of each frame, while "×" represents a pixel belonging to the second field of each frame.

Suppose the first field of the frame **n+2** should be converted into a progressive picture (i.e., progressively scanned picture) with a twice higher resolution. And it will be

described how to generate the pixel **D** (represented by "△") shown in FIG. **10**.

In this case, a motion vector **A** for an area **G**, including the pixel **D** to be generated, is used among the motion vectors **MV** output from the video decoder **110**. The motion vector **A** originates from an area **F** belonging to the first field of the frame **n**. And a motion vector **E** with the same direction as that of the motion vector **A** is selected from the motion vectors of the field closest to the pixel **D**, i.e., the second field of the frame **n+1**.

Then, a more accurate motion vector **B** for a pixel neighboring the pixel **D** is selected from motion vectors with directions similar to that of the motion vector **E** by using the pixels neighboring the pixel **D** in the first field of the frame **n+2** and the pixels in the second field of the frame **n+1**. That is to say, the pixel **D** can be regarded as a pixel **C** that has moved from the second field of the frame **n+1** by the motion vector **B**.

The motion vector estimating section **121a** outputs the motion vector **B**, which has been estimated in this manner, as the pixel-by-pixel motion vector **MV**α to the motion recognizing section **121b**.

Using the motion vector **MV**α that has been input from the motion vector estimating section **121a**, the motion recognizing section **121b** distinguishes the quasi-still and moving

areas from each other. In the illustrated embodiment, pixels, for which no pixel-by-pixel motion vectors **MV** α have been found by the motion vector estimating section **121a**, are regarded as belonging to the moving area. On the other hand, pixels, for which the pixel-by-pixel motion vectors **MV**α have been found, are regarded as belonging to the quasi-still area. The motion recognizing section **121b** performs the motion recognition in this manner, and passes the result to the picture dividing section **121c**.

As used herein, the "quasi-still area" means an area of a picture that can be handled as if it were a still picture by shifting the picture by the quantity of the motion using the pixel-by-pixel motion vector **MV**α although that area is actually a part of the picture with motion.

In accordance with the result of motion recognition obtained from the motion recognizing section **121b**, the picture dividing section **121c** divides a frame picture into the quasi-still and moving areas. Part of the video data that has been classified as the quasi-still area is input to the quasi-still-area resolution converting section **123**, while the other part of the video data classified as the moving area is input to the moving-area resolution converting section **124**. The quasi-still-area and moving-area resolution converting sections **123** and **124** perform resolution conversion on the input video data by respective methods.

Hereinafter, it will be described with reference to FIG. **10** how the quasi-still-area resolution converting section **123** performs the resolution conversion. The quasi-still-area resolution converting section **123** performs the resolution conversion using the original video signal **SLR**, the moving area picture output from the area dividing section **121** and the pixel-by-pixel motion vector **MV**α output from the motion vector estimating section **121a**.

Suppose the first field of the frame **n+2** should be subjected to the resolution conversion, for example. In this case, since the motion vector **B** has been estimated as the pixel-by-pixel motion vector **MV**α for a pixel neighboring the pixel **D**, the pixel **C** belonging to the second field of the frame **n+1** is retrieved from the frame memory **127**. And the pixel value thereof is used as the value of the pixel **D**. In the same way, pixels at other interpolation locations are also generated using the pixel-by-pixel motion vectors **MV** α that have been estimated by the motion vector estimating section **121a**. As a result, the high-resolution video signal is generated.

Hereinafter, it will be described with reference to FIG. **11** how the moving-area resolution converting section **124** performs the resolution conversion. In FIG. **11**, "○" represents a pixel for the low-resolution video signal, while "×" represents a pixel generated by interpolation for the high-

resolution video signal. As shown in FIG. 11, the moving-area resolution converting section **124** generates the high-resolution video signal from the pixels in the field of the low-resolution video signal through interpolation. For example, the value of a pixel **C** for the high-resolution video signal is the same as that of a pixel **A** for the low-resolution video signal. The value of a pixel **D** for the high-resolution video signal is obtained from those of the pixels **A** and **B** for the low-resolution video signal.

The video signals, which have been generated by the quasi-still-area and moving-area resolution converting sections **123** and **124**, are input to the area combining section **125**. The area combining section **125** combines the input video signals together, thereby outputting the combined signal as a new video signal **SHR**.

This embodiment includes the motion vector estimating section **121a** for re-calculating the motion vector **MV**α on a pixel-by-pixel basis using the motion vector **MV** that has been obtained from the encoded stream **CS** of the original video signal **SLR**. Alternatively, the area division may also be carried out using the value of the estimated motion vector **MV** as it is.

FIG. **12** is a diagram illustrating a configuration for a resolution converter **120A** for such an alternative embodiment. In FIG. **12**, the area dividing section **122** includes a motion

recognizing section **122a** for recognizing the motion based on the motion vector **MV** directly and a picture dividing section **122b**. In this case, the area division may be carried out with a pixel regarded as belonging to the quasi-still area if the absolute value of the motion vector **MV** is less than a predetermined threshold value or as belonging to the moving area if the absolute value is greater than the threshold value, for example. Also, a pixel may be regarded as belonging to the moving area only when the absolute value of the motion vector **MV** is very great. Furthermore, the resolution conversion may be carried out while regarding the entire area as the quasi-still area and using the value of the motion vector **MV** as it is.

As described above, according to this embodiment, an encoded stream of an original video signal is decoded while a motion vector is extracted from the encoding stream. Then, a new video signal is generated through resolution conversion using the motion vector extracted. In this manner, the amount of processing, including motion vector calculation for resolution conversion purposes, can be reduced considerably. In addition, quasi-still and moving areas of a picture represented by the original video signal are distinguished by the motion vector and the resolution conversion is carried out on the quasi-still area using the motion vector. Thus, the amount of processing, including motion vector calculation carried out to

convert the resolution for the quasi-still area, can be cut down significantly. Furthermore, the resolution conversion is performed with mutually different methods applied to the quasi-still and moving areas. As a result, a new video signal of quality can be obtained with the amount of processing drastically reduced.

In this embodiment, an original video signal with a low resolution is converted into a new video signal with a high resolution. However, even if a high-resolution original video signal should be converted into a low-resolution new video signal conversely, the motion vectors may also be used as in this embodiment.

Also, in this embodiment, the motion vector section **121a** estimates a motion vector between a current field and a past field, and the quasi-still-area resolution converting section **123** generates an interpolated pixel from pixels belonging to a preceding field. Alternatively, a motion vector may also be estimated for a future field and an interpolated pixel may also be generated from pixels belonging to a future field.

Moreover, in this embodiment, the moving-area resolution converting section **124** converts the resolution by generating interpolated pixels as shown in FIG. **11**. Optionally, any other resolution conversion method is applicable.

Furthermore, in this embodiment, the quasi-still-area resolution converting section **123** generates an interpolated pix-

el from pixels belonging to the closest, preceding field. Alternatively, an interpolated pixel may also be generated from pixels belonging to a field that is several fields away from the current field.

(Embodiment 3)

FIG. **13** is a block diagram illustrating an overall configuration for an image processing apparatus according to a third embodiment of the present invention. As shown in FIG. **13**, the image processing apparatus according to this embodiment includes video decoder **210**, resolution converter **220**, video encoder **230** and encoding parameter changer **240**.

A first encoded stream **CSA**, which is an encoded stream of an original video signal with a high resolution (i.e., a first resolution), is input. In response, the video decoder **210** decodes the input first encoded stream **CSA** into the original video signal **SHR**. Not just this decoding, the decoder **210** also extracts an encoding parameter **PARA** from the first encoded stream **CSA**. The resolution converter **220** converts the original video signal **SHR** into a new video signal **SLR**. The encoding parameter changer **240** changes the first encoding parameter **PARA**, which has been output from the video decoder **210**, into a parameter for a video with a low resolution and then outputs it as a second encoding parameter **PARB**. And the video encoder **230** encodes the new video signal **SLR** us-

ing the second encoding parameter **PARB** and then outputs it as a second encoded stream **CSB**.

The first encoded stream **CSA** is herein supposed to have been encoded in compliance with the MPEG-2. Accordingly, the video decoder **210** is a decoder complying with the MPEG-2 standard. The configuration and operation of the video decoder **210** are the same as those of the video decoder **10** according to the first embodiment. Thus, the detailed description thereof will be omitted herein.

The resolution converter **220** converts the original video signal **SHR**, which has been decoded by the video decoder **210**, into the new video signal **SLR**. In the illustrated embodiment, the resolution converter **220** cuts down the number of pixels by half both vertically and horizontally.

Next, it will be described with reference to FIG. **14** how the encoding parameter changer **240** operates. The encoding parameter changer **240** changes the first encoding parameter **PARA** for the original video signal **SHR** into the second encoding parameter **PARB** for the new video signal **SLR**. Suppose a frame picture of the original video signal **SHR** is converted into a frame picture of the new video signal **SLR** that has had its number of pixels reduced by half both vertically and horizontally as shown in FIG. **14(a)**.

First, a situation where the encoding parameter to be changed is a motion vector will be described with reference to

FIG. **14(b)**. In FIG. **14(b)**, a frame picture is illustrated as having been divided into multiple macroblocks. As shown in FIG. **14(b)**, an area for macroblocks **MBH1**, **MBH2**, **MBH3** and **MBH4** in the original video signal **SHR** (i.e., the first area) is downscaled to an area for a macroblock **MBL** in the new video signal **SLR** (i.e., the second area) through the resolution conversion. In this case, a motion vector **MVL** for the macroblock **MBL** in the new video signal **SLR** is supposed to be obtained by averaging weighted motion vectors **MVHi** of the respective macroblocks **MBHi** in the original video signal **SHR**. That is to say, the motion vector **MVL** is given by the following Equation (1):

$$MVL = \left\{ \sum_{i=1}^{N} (Ci \times MVHi) \right\} \times \alpha \qquad (1)$$

where Ci is a weighting coefficient meeting the following Equation (2):

$$\sum_{i=1}^{N} Ci = 1 \qquad (2)$$

where N is the number of macroblocks in the original video signal **SHR** that are used to estimate the motion vector **MVL** for the macroblock **MBL** in the new video signal **SLR**. In the illustrated example, N is "4". Also, α is a ratio of the number of pixels representing the picture size of the new video sig-

nal to the number of pixels representing the picture size of the original video signal. Normally, a horizontal α value is different from a vertical α value.

In this case, if the pixel number ratio resulting from the resolution conversion is 1/2 or 1/4, for example, then the boundary between macroblocks after the conversion matches to the boundary between macroblocks before the conversion. Accordingly, the same weighting coefficient **Ci** value can be used. However, if the pixel number ratio is 1/3 or 2/5, for example, then the boundary between macroblocks after the conversion does not match to the boundary between macroblocks before the conversion. Thus, the macroblocks should be weighted by mutually different weighting coefficient **Ci** values before the conversion.

Next, a situation where the encoding parameter to be changed is a DCT type (i.e., type of orthogonal transformation) will be described with reference to FIG. **14(c)**. In FIG. **14(c)**, a frame picture is illustrated as having been divided into multiple blocks. As shown in FIG. **14(c)**, an area for blocks **BH1**, **BH2**, **BH3** and **BH4** in the original video signal (i.e., the first area) is downscaled to an area for a block **BL** in the new video signal (i.e., the second area) through the resolution conversion. In this case, a DCT type for the block **BL** in the new video signal is obtained by using the DCT types of the respective blocks **BH1** through **BH4** in the original video

signal. For example, if the DCT types of the blocks **BH1** through **BH4** are all a frame structure, then the DCT type of the block **BL** may also be the frame structure. Or if at least one of the DCT types of the blocks **BH1** through **BH4** is a field structure, then the DCT type of the block **BL** should also be the field structure. Alternatively, the DCT type of the converted block may be defined as the DCT type applied to the greater number of original blocks.

In this manner, the encoding parameter changer **240** changes the first encoding parameter **PARA** for the original video signal **SHR** into the second encoding parameter **PARB** for the new video signal **SLR**.

FIG. **15** is a block diagram illustrating an internal configuration for the video encoder **230**. The video encoder **230** shown in FIG. **15** has basically the same configuration as a normal MPEG encoder. However, since a motion vector may be provided as the second encoding parameter **PARB**, the motion estimating section is omitted therefrom. Also, since a DCT type may be provided as the second encoding parameter **PARB**, means for determining the DCT type by calculating a variance, for example, is omitted from a DCT calculating section **233**.

In the illustrated example, the video encoder **230** is supposed to encode the new video signal **SLR** in compliance with the MPEG-2 standard, and is also supposed to perform inter-frame encoding.

First, the new video signal **SLR**, which has been input to the video encoder **230**, is divided by a blocking section **231** into multiple macroblocks (each consisting of 16×16 pixels). Then, the macroblocks are sequentially input to a motion compensation section **232** one after another. The motion compensation section **232** performs motion compensation on the input macroblocks by using the motion vector included in the second encoding parameter **PARB** that has been provided from the encoding parameter changer **240**. Specifically, the motion compensation section **232** reads out a reference macroblock from a frame memory **238** by using the motion vector and obtains a difference between the reference and input macroblocks, thereby performing the motion compensation. The resultant differential macroblock is input to the DCT calculating section **233**.

The DCT calculating section **233** converts the differential macroblock into DCT coefficients on an 8×8 pixel block basis. In this case, the DCT calculating section **233** performs the DCT calculation in accordance with the DCT type included in the second encoding parameter **PARB** that has been provided from the encoding parameter changer **240**. The resultant DCT coefficients are output to a quantizing section **234**, which performs a quantization process on the DCT coefficients. A variable-length encoding section **235** performs a variable-length encoding process on the output of the quantizing section **234** and then outputs the result as the second encoded

stream **CSB**.

Also, the output of the quantizing section **234** is decoded by an inverse quantizing section **236** and an inverse DCT section **237**. Subsequently, the decoded data is added to the reference macroblock, which has been read out from the frame memory **238**, by an adder **239**, and then the sum is stored on the frame memory **238**. The data stored will be used as a reference picture when succeeding frames are encoded.

As described above, according to this embodiment, a first encoded stream of an original video signal is decoded and the decoded original video signal is converted into a new video signal through resolution conversion. Also, in parallel with this decoding, a first encoding parameter such as motion vector or DCT type is extracted from the first encoded stream. Next, the first encoding parameter is changed into a second encoding parameter for the new video signal, and the new video signal is converted into a second encoded stream by using this second encoding parameter. Accordingly, there is no need to obtain encoding parameters in encoding the new video signal, thus cutting down the amount of processing considerably. Particularly when motion vectors are used as the encoding parameters, the amount of processing can be reduced very significantly.

In this embodiment, the encoding parameter to be changed is a motion vector or a DCT type. However, the present inven-

tion is implementable similarly even by using any other encoding parameter such as a mode of motion compensation.

Also, the encoding parameter changing method of this embodiment is just an illustrative one. Thus, the conversion may be carried out by any other method. In this embodiment, the encoding parameter for a predetermined area (i.e., macroblock or block) of the new video signal is obtained by using only the encoding parameter for an area of the original video signal that corresponds to the predetermined area. Optionally, not just the encoding parameter for the area of the original video signal corresponding to the predetermined area, but also encoding parameters for other surrounding areas may be used.

Moreover, in this embodiment, the motion vectors are converted by adopting a weighting and averaging method as a predefined arithmetic operation. Examples of other imaginable and applicable methods include a method of selecting a centroid of a motion vector component, a method of neglecting small motion vector quantities and a method of weighting using predicted errors.

Furthermore, in this embodiment, the same weighting coefficient Ci is used if the macroblock boundaries match before and after the conversion. However, the coefficient does not always have to be the same.

Furthermore, in this embodiment, an original video sig-

nal with a high resolution is converted into a new video signal with a low resolution. However, even if a low-resolution original video signal should be converted into a high-resolution new video signal conversely, the encoding parameters may also be changed as in this embodiment.

(Embodiment 4)

FIG. **16** is a block diagram illustrating an overall configuration for an image processing apparatus according to a fourth embodiment of the present invention. As shown in FIG. **16**, the image processing apparatus according to this embodiment includes video decoder **310**, resolution converter **320**, video encoder **330** and motion compensation setter **340**.

A first encoded stream **CSA**, which is an encoded stream of an original video signal with a high resolution (i.e., a first resolution), is input. In response, the video decoder **310** decodes the input first encoded stream **CSA** into the original video signal **SHR**. Not just this decoding, the decoder **310** also extracts a motion vector **MV** from the first encoded stream **CSA**. The resolution converter **320** converts the original video signal **SHR** into a new video signal **SLR** with a low resolution (i.e., a second resolution). The motion compensation setter **340** defines settings for the motion compensation, which should be carried out by the video encoder **330**, by using the motion vector **MV** of the encoded stream **CSA** that

has been output from the video decoder **310**. And the video encoder **330** estimates a second motion vector in accordance with the setting information **SET** output from the motion compensation setter **340**, encodes the new video signal **SLR** using the second motion vector and then outputs it as a second encoded stream **CSB**.

The configurations and operations of the video decoder **310** and resolution converter **320** are the same as those of the video decoder **110** and resolution converter **320** according to the second embodiment. Thus, the detailed description thereof will be omitted herein.

Hereinafter, it will be described how the motion compensation setter **340** operates. The motion compensation setter **340** receives the motion vector **MV** that has been obtained by the video decoder **310** for the original video signal **SHR**, and obtains the setting information **SET** for motion compensation to be carried out by the video encoder **330**. As the setting information **SET**, an initial value of motion compensation (i.e., an initial value of the second motion vector) and a range of motion compensation (i.e., a search range for the second motion vector) are determined, for example.

First, it will be described with reference to FIG. **14(b)** how the initial value of motion compensation is obtained as the motion compensation setting information **SET**. In FIG. **14(b)**, a frame picture is illustrated as having been divided

into multiple macroblocks. As shown in FIG. **14(b)**, an area for macroblocks **MBH1**, **MBH2**, **MBH3** and **MBH4** in the original video signal is downscaled to an area for a macroblock **MBL** in the new video signal through the resolution conversion. In this case, the motion compensation setter **340** may obtain the motion vector **MVL** by the Equation (1) described above, for example, as a motion compensation initial value that will be used by the video encoder **310** in encoding the macroblock **MBL**. And the setter **340** outputs this motion vector **MVL** as the motion compensation setting information **SET**.

Next, it will be described with reference to FIG. **14(b)** how the range of motion compensation is determined as the motion compensation setting information **SET**. In that case, the motion compensation setter **340** determines the motion compensation range, from which the video encoder **330** will have to estimate a motion vector for the macroblock **MBL**, by using the motion vectors of the macroblocks **MBH1** through **MBH4**. To define the motion compensation range, the maximum absolute values of positive and negative motion vector components of the respective macroblocks **MVH1** through **MVH4** may be used according to a method, for example. Alternatively, according to another method, the average or variance value of the motion vectors of the respective macroblocks **MVH1** through **MVH4** may also be used.

In this manner, the motion compensation setter **340** defines the initial value or range for calculating the motion

vector for the new video signal as the motion compensation setting information **SET** by using the motion vector **MV** of the original video signal **SHR**. Then, the setter **340** outputs the value or range to the video encoder **330**.

FIG. **17** is a block diagram illustrating an internal configuration for the video encoder **33**. In the illustrated example, the video encoder **330** is supposed to encode the new video signal **SLR** in compliance with the MPEG-2 standard, and is also supposed to perform inter-frame encoding.

First, the new video signal **SLR**, which has been input to the video encoder **330**, is divided by a blocking section **331** into multiple macroblocks (each consisting of 16×16 pixels). Then, the macroblocks are sequentially input to a motion compensation section **332** and a motion vector calculator **350** one after another.

The motion vector calculator **350** calculates the second motion vector **MV2** for each of the input macroblocks in accordance with the motion compensation setting information **SET** that has been provided from the motion compensation setter **340**. At this stage, a reference picture is retrieved from a frame memory **338**.

Hereinafter, it will be described with reference to FIG. **18** how the motion vector calculator **350** operates. In FIG. **18**, **(a)** illustrates a frame to be encoded, which is represented by the new video signal **SLR** and has been input from

the resolution converter **320**, while **(b)** and **(c)** illustrate the reference frames that have been read out from the frame memory **338**. Suppose a motion vector should be estimated for the macroblock **MB1** shown in FIG. **18(a)**.

First, a situation where an initial value of motion compensation has been input as the motion compensation setting information **SET** will be described with reference to FIG. **18(b)**. In this case, the motion vector calculator **350** uses the initial value **SET**, which has been obtained by the motion compensation setter **340**, as the initial value of the second motion vector **MV2**. And an area surrounding the endpoint of the motion vector **SET** originating from a macroblock **MB1** α, which is located at the same position as the macroblock **MB1**, is defined as a search range **SR1** in which the second motion vector **MV2** should be estimated. In this search range **SR1**, the second motion vector **MV2** is calculated for the macroblock **MB1**.

Next, a situation where a range of motion compensation has been input as the motion compensation setting information **SET** will be described with reference to FIG. **18(c)**. In that case, the motion vector calculator **350** uses the range **SET**, which has been obtained by the motion compensation setter **340**, as the search range **SR2** in which the second motion vector **MV2** should be estimated. And in this search range **SR2** (i.e., the range **SET**), the second motion vector **MV2** is calculated for the macroblock **MB1**.

The motion compensation section **332** performs motion compensation on the input macroblocks by using the second motion vector **MV2** provided from the motion vector calculator **350**. Specifically, the motion compensation section **332** reads out a reference macroblock from the frame memory **338** by using the second motion vector **MV2** and obtains a difference between the reference and input macroblocks, thereby performing the motion compensation. The resultant differential macroblock is input to a DCT calculating section **333**.

The DCT calculating section **333**, quantizing section **334**, variable-length encoding section **335**, inverse quantizing section **336**, inverse DCT calculating section **337** and adder **339** operate in the same way as the counterparts of the second embodiment. Thus, the description thereof will be omitted herein. The second encoded stream **CSB** is output from the variable-length encoding section **335**.

As described above, according to this embodiment, a first encoded stream of an original video signal is decoded and the decoded original video signal is converted into a new video signal through resolution conversion. Also, in parallel with this decoding, a first motion vector that has been used for encoding the original video signal is extracted. Next, motion compensation setting information such as the initial value and range is defined to estimate a second motion vector, which will be used for encoding the new video signal, from the

first motion vector. And in accordance with this setting information, the second motion vector is estimated, thereby encoding the new video signal. As a result, the search range can be much narrowed compared to the prior art, and the amount of motion vector computation can be considerably reduced in encoding a new video signal. Also, since the motion compensation setting information can be obtained from the first motion vector that has been used for encoding the original video signal, the accuracy of motion compensation can be kept high.

Also, the method of defining the motion compensation setting information according to this embodiment is just an illustrative one. Thus, any other method may be used. For example, not just the motion vector for a macroblock of the original video signal corresponding to a predetermined macroblock of the new video signal, but also motion vectors for other macroblocks surrounding the former macroblock may be used. Moreover, in this embodiment, the initial value of the second motion vector is determined by adopting a weighting and averaging method. Examples of other imaginable and applicable methods include a method of selecting a centroid of a motion vector component, a method of neglecting small motion vector quantities and a method of weighting using predicted errors.

Furthermore, the motion compensation setter **340** may define both the initial value and search range of the second

motion vector as the motion compensation setting information **SET**.

Furthermore, in this embodiment, an original video signal with a high resolution is converted into a new video signal with a low resolution. However, even if a low-resolution original video signal should be converted into a high-resolution new video signal conversely, the motion compensation setting information **SET** may also be defined as in the present invention.

(Embodiment 5)

FIG. **19** is a block diagram illustrating an overall configuration for an image processing apparatus according to a fifth embodiment of the present invention. As shown in FIG. **19**, the image coding apparatus according to this embodiment includes video decoder **410**, resolution converter **420**, video encoder **430** and system control section **440**.

An encoded stream **CSA** of a video signal with a high resolution (i.e., a first resolution) is input. In response, the video decoder **410** decodes the input encoded stream **CSA** into the original video signal **SHR**. The resolution converter **420** converts the decoded original video signal **SHR** into a new video signal **SLR** with a low resolution (i.e., a second resolution). The video encoder **430** encodes the new video signal **SLR** and then outputs it as a second encoded stream **CSB**.

The first encoded stream **CSA** is herein supposed to have been encoded in compliance with the MPEG-2. Accordingly, the video decoder **410** is a decoder complying with the MPEG-2 standard. The configuration and operation of the video decoder **410** are the same as those of the video decoder **10** according to the first embodiment. Thus, the detailed description thereof will be omitted herein.

As shown in FIG. **20**, the resolution converter **420** converts the original video signal **SHR** into a video signal representing a letterbox picture. In the example illustrated in FIG. **20**, the original video signal **SHR** with an aspect ratio of 16:9 as shown in **(a)** is converted into a new video signal **SLR** with an aspect ratio of 4:3 as shown in **(b)**. If the picture has been converted into the letterbox picture, black-level areas **451** and **452** in the shape of bands are added to the frame picture **450** of the new video signal **SLR** over and under the picture.

FIG. **21** is a block diagram illustrating an internal configuration for the video encoder **430**. As shown in FIG. **21**, the video encoder **430** includes area dividing section **431**, blocking section **432**, DCT calculating section **433**, quantizing section **434**, variable-length encoding section **435**, encoded stream generating section **436** and encoded stream storage section **437**. In the illustrated embodiment, the video encoder **430** is supposed to encode the new video signal **SLR** by the

MPEG-2 and is also supposed to perform intra-frame encoding.

On receiving an instruction signal **SI** from the system control section **440**, the area dividing section **431** extracts and outputs an effective data area (i.e., the area **450** except the black-level areas **451** and **452** shown in FIG. **20(b)**) from the letterbox picture represented by the new video signal **SLR**. That is to say, the video signal representing the black-level areas **451** and **452** is not output. The blocking section **432** divides the video signal representing the input area **450** into multiple blocks. The DCT calculating section **433** converts the blocked video signal into DCT coefficients. The DCT coefficients, output from the DCT calculating section **433**, are quantized by the quantizing section **434**, converted by the variable-length encoding section **435** into a first encoded stream **CSB1** and then output to the encoded stream generating section **436**.

On the encoded stream storage section **437**, second encoded streams **CSB2**, which have been obtained by encoding the video signal representing the black-level areas **451** and **452**, are stored in advance. Both intra-frame and inter-frame encoded streams are stored as the second encoded streams **CSB2** on the encoded stream storage section **437**. In the illustrated embodiment, one of the second encoded streams **CSB2** that has been intra-frame encoded is output from the encoded stream storage section **437**.

The encoded stream generating section **436** combines the first encoded stream **CSB1** output from the variable-length encoding section **435** with the second encoded stream **CSB2** output from the encoded stream storage section **437**. In the illustrated embodiment, these encoded streams are combined sequentially by beginning with the top of the frame. That is to say, the encoded stream representing the black-level area **451** output from the encoded stream storage section **437**, the encoded stream representing the area **450** output from the variable-length encoding section **435** and the encoded stream representing the black-level area **452** output from the encoded stream storage section **437** are combined together in this order. The encoded stream combined in this manner is output as the encoded stream **CSB**.

It should be noted that the resolution converter **420** may output the new video signal **SLR** with the black-level areas **451** and **452** eliminated. In that case, the area dividing section **431** may be omitted from the video encoder **430**.

Also, the area dividing section **431** may determine whether the input new video signal **SLR** represents a letterbox picture or not. In that case, there is no need to externally provide the instruction signal **SI** to the video encoder **430**.

As described above, according to this embodiment, where a new video signal with a letterbox structure, obtained from an original video signal through resolution conversion, is en-

coded, no encoding is actually performed on black-level areas thereof but pre-stored encoded streams are combined. Accordingly, there is no need to perform encoding processing on the black-level areas, thus considerably cutting down on the amount of processing. For example, suppose a frame represented by the new video signal has a size of 720 horizontal pixels by 480 vertical pixels and the area other than the black-level areas has a vertical size of 360 pixels. In that case, the amount of processing can be reduced by 25% compared to encoding the entire picture.

In this embodiment, the black-level areas are added over and under the effective data area. Alternatively, at least one black-level area may be added either over or under the data area. Also, the black-level area(s) may be added to various parts other than the upper and lower parts, e.g., on right- and left-hand sides, on the left side only or on the right side only.

Moreover, in this embodiment, the original video signal with an aspect ratio of 16:9 is converted into the new video signal with an aspect ratio of 4:3. However, aspect ratios in any other combination may be used instead.

Furthermore, in this embodiment, an original video signal with a high resolution is converted into a new video signal with a low resolution. However, even if a low-resolution original video signal should be converted into a high-

resolution new video signal, encoding of the black-level areas can also be omitted as in the present invention.

Furthermore, in this embodiment, the video encoder **430** performs intra-frame encoding. But the same statements are applicable to a situation where the video encoder **430** performs inter-frame encoding.

Furthermore, in the foregoing embodiments, the MPEG-2 is used as an exemplary encoding method. Alternatively, any other encoding method like the MPEG-1 or the H.261 is also applicable.